PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 17/60 — A1

(11) International Publication Number: **WO 99/03053**

(43) International Publication Date: 21 January 1999 (21.01.99)

(21) International Application Number: PCT/US98/13585

(22) International Filing Date: 7 July 1998 (07.07.98)

(30) Priority Data:
08/891,129 10 July 1997 (10.07.97) US

(71) Applicant: EMSEMBLE SOLUTIONS, INC. [US/US]; Suite 250, 4675 Stevens Creek Boulevard, Santa Clara, CA 95051 (US).

(72) Inventors: HORSTMANN, Jens, U.; 3633 Vireo Avenue, Santa Clara, CA 95051 (US). SPIES, Frank, M.; Apartment 127, 1600 Villa Street, Mountain View, CA 94041 (US).

(74) Agents: OGONOWSKY, Brian, D. et al.; Skjerven, Morrill, MacPherson, Franklin & Friel LLP, Suite 700, 25 Metro Drive, San Jose, CA 95110 (US).

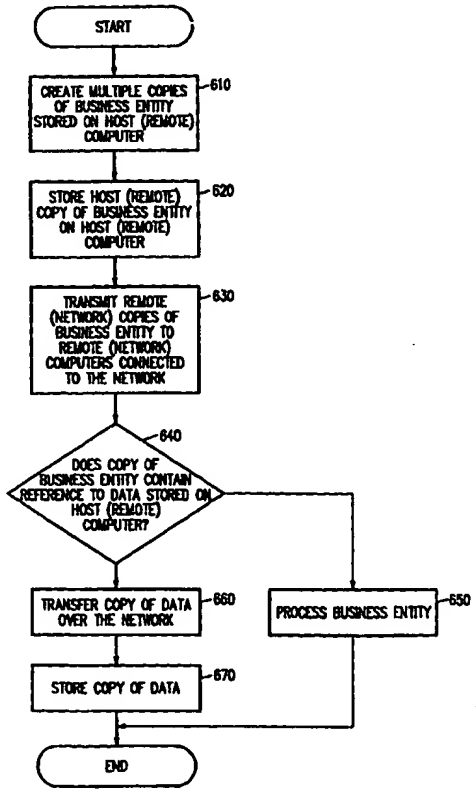
(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, GW, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: AUTOMATED PROCESSING OF BUSINESS ENTITIES DISTRIBUTED OVER A NETWORK

(57) Abstract

A method and a system are provided for processing business entities over a computer network. The system is not disabled by a disruption of communications or by heavy network traffic conditions on the computer network. The method and system of the present invention allow for extensive collaboration among distributed business entities and even for automatic modification of the functionality of distributed business entities. The method initially creates multiple copies of a business entity on a host computer connected to the network. One of the copies of the business entity is then stored on the host compouter, while the others are transmitted to remote computers over the network. When an operation (e.g., making a reservation) is performed on a copy of the business entity (regardless of which computer the copy of the business entity is stored on), the remaining copies of the business entity are automatically updated. Business entities can operate in an event-driven fashion (i.e., responding to a change in the business attributes or the business rules of one or more copies of a business entity). Thus, the functionality of all copies of a business entity can be modified by a change of a single copy of the business entity. Since business entities may collaborate to perform a task, the operation of a business entity may also be driven by a change in a copy of a different business entity. Finally, distributed copies of a business entity may be further distributed to additional computers.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

AUTOMATED PROCESSING OF BUSINESS ENTITIES DISTRIBUTED OVER A NETWORK

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention is related generally to a computer system and, in particular, to a method and apparatus for processing business entities distributed over a network.

Related Art

Current systems for conducting electronic commerce over a computer network (e.g., the Internet) typically employ a client-server model for requesting and obtaining information using computers connected to the network. For example, a travel agency trying to sell vacation packages on the Internet typically maintains a web page that provides information about the services offered to customers by the agency. The web page may include one or more links to other web pages. For example, links to web pages maintained by individual travel service providers (e.g., airlines, hotels and car rental companies) are typically included in travel agencies' web pages. Customers of the travel agency can thus book vacation packages either through the agency's web page or by following links to individual airlines, hotels and car rental companies' web pages.

In addition, some prior art electronic commerce systems use distributed objects, in addition to web pages, to provide more extended functionality on the client side. One such system [hereinafter the "ClubMed system"] is described on pages 541-607 of "Client Server Programming with JAVA™ and CORBA" by

Robert Orfali and Dan Harkey (John Wiley and Sons : New York, NY, 1997). The structure of the ClubMed system is shown in Fig. 1.

In Fig. 1, several clients 100 communicate with a server 110 via a network 120. Objects are distributed between the server 110 and the clients 100 via CORBA/IIOP modules 140. Information extracted from database 130 is organized in several objects 150 stored on server 110. The objects 150, in turn, are distributed to clients 100 via CORBA/IIOP modules 140 over the network 120. Finally, applets 160 are executed by clients 100 to display the information of objects 150.

A user of client 100 can access a ClubMed web page from server 110, check on prices and availability of resorts during different seasons and make or cancel reservations, directly from client 100. While the ClubMed system represents an improvement over the simple web page reservation system described above, the ClubMed system still employs a client-server model. For example, when the user of client 100 requests a reservation, the request is sent to server 110 which processes the request and returns a confirmation number back to the client 100 and the user. Thus, if either the network connection between client 100 and server 110 is disrupted or server 110 is unavailable, the ClubMed system is disabled. Furthermore, during periods of heavy network traffic on network 120, the user of client 100 may experience lengthy time delays in making or canceling the reservation.

As a result, there is a need for a method and a system to process business entities over a network that is not disabled by the interruption of a network connection between computers connected to the network and whose performance is not disrupted by periods of heavy network traffic over the network.

SUMMARY OF THE INVENTION

Accordingly, a method and a system are provided for processing business entities over a computer network. The system is not disabled by a disruption of

communications over the computer network and the system's performance is not significantly affected by heavy network traffic conditions on the computer network. The method and system of the present invention allow for extensive collaboration among distributed business entities and even for automatic modification of the functionality of distributed business entities.

The method initially creates multiple copies of a business entity on a host computer connected to the network. One of the copies of the business entity is then stored on the host computer, while the others are transmitted to remote computers over the network. When an operation (e.g., making a reservation) is performed on a copy of the business entity (regardless of which computer the copy of the business entity is stored on), the remaining copies of the business entity are automatically updated. Since business entities have both business attributes and business rules, the functionality of all copies of the business entity can be altered as a result of an operation performed on a single copy of the business entity.

The system is not disabled by a disruption of communications over the computer network (or by heavy traffic conditions on the computer network) after the copies of the business entity have been distributed, since each copy of the business entity can be operated upon independently. The remaining copies of the business entity can then be updated to maintain consistency when the network connection is reestablished. As a result, a user of the system does not have to wait for a response from a server before completing the operation, as required by prior art systems.

Furthermore, business entities can operate in an event-driven fashion (i.e., responding to a change in the business attributes or the business rules of one or more copies of a business entity). Thus, the functionality of all copies of a business entity can be modified by a change in the business attributes of a single copy of the business entity. Since business entities may collaborate to perform a task, the operation of a business entity may also be driven by a change in a copy of a different business entity.

Finally, distributed copies of a business entity may be further distributed to additional computers.

BRIEF DESCRIPTION OF THE DRAWING

Fig. 1 illustrates a prior art electronic commerce system using distributed objects.

Fig. 2 illustrates a business entity processing system according to one embodiment of the present invention.

Fig. 3 is a diagram of the business entity classes of a travel reservation system, according to one embodiment of the invention.

Fig. 4A illustrates a copy of business entity A on computer 210 of Fig. 2.

Fig. 4B illustrates a copy of business entity A on computer 240 of Fig. 2.

Fig. 5 is a flow diagram of an automated update of business entities operation, according to one embodiment of the invention.

Fig. 6 is a flow diagram of a data caching operation, according to one embodiment of the invention.

Fig. 7 is a block diagram of a computer system according to one embodiment of the invention.

Fig. 8A is a block diagram of a security system, according to one embodiment of the invention.

Fig. 8B shows a firewall, according to one embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

A system for automated processing of business entities, according to one embodiment of the invention, is shown in Fig. 2. The system has several computers 210, 220, 230, 240 and 250 connected to a computer network 200. Each of computers 210, 220, 230, 240 and 250 has an adapter (not shown in the drawing) that allows the computer to process business entities distributed to other computers over the network. The adapter is implemented as a computer program

that handles transfers of business entities among computers connected to the network, as well as providing additional services such as data caching and automated synchronization of business entities. Data caching is a process by which portions of data stored on a host computer connected to a network are transmitted to and stored on a remote computer to allow faster access to the data by the remote computer. Automated synchronization is a process by which portions of the data stored on more than one computer connected to the network (e.g., for data caching purposes) are updated to ensure data consistency across different computers. As those skilled in the art are familiar with these processes, they are not further described herein.

In addition, computers 210, 220 and 230 have databases, 215, 225 and 235. Legacy gateways 217, 227 and 237 are used to interface between computers 210, 220 and 230 and databases 215, 225 and 235. Legacy gateways 217, 227 and 237 translate data formatted for processing by legacy applications to and from a format used in processing business entities. Computers 210, 220 and 230 have copies of business entities A, B and C, respectively. Copies of business entities A, B and C can be transferred between computers 210, 220, 230, 240 and 250 via computer network 200. Thus data stored in databases 215, 225 and 235 can be converted into copies of business entities A, B and C and transferred between computers connected to the network. Copies of business entities A, B and C can then be processed independently on each of the computers to which they are transferred.

Once a copy of a business entity is modified on one of the computers, the other copies of the business entity are automatically updated to ensure data consistency. However, unlike in prior art systems where data was updated synchronously as part of the operation modifying the data, the embodiments of the invention allow for updates to be performed asynchronously, after the operation modifying the business entity is completed. As a result, the functioning of a system according to an embodiment of the invention is not interrupted by a

disruption of communications over the network nor is it significantly impacted by heavy traffic conditions over the network.

This is accomplished by incorporating both business attributes and business rules into business entities so that business entities transmitted from a host computer to a remote computer over the network are able to perform on the remote computer any operations which could be performed on the host computer without the need for further communications with the host computer. For example, the ClubMed system used in describing the prior art can be implemented using an automated business processing system, according to an embodiment of the invention, in the following manner. First, a number of business entities are created to model a reservation system of a vacation resort. Each business entity includes both business attributes and business rules. Business attributes are data structures for storing information concerning the item represented by the business attribute. A data structure is a set of data elements (variables, pointers, data fields, links, images, etc.) having a relationship defined by the data structure. Examples of data structures are vector arrays, constructs, linked lists, records, etc. Business rules are operations that can be performed on business attributes. Business rules can be either functions (i.e., segments of computer code that can be invoked through an API) or rules (i.e., segments of computer code that are automatically invoked when certain data elements are updated). Finally, business rules are tied to business attributes since each business rule is designed to operate on certain business attributes.

According to the object oriented programming paradigm, business entities are organized in classes arranged in a hierarchical order, so that common properties of business entities may be shared among different classes by inheritance. Thus, both business attributes (data structures) and business rules (code) can be shared via inheritance among business entities.

Fig. 3 illustrates business entity classes that could be used to implement a reservation system similar to the ClubMed example described with respect to prior

art systems. Each of the business entity classes of Fig. 3 is represented by a bubble containing the name of the business entity class in bold on the top line, followed by two columns of text listed side by side under the name. The left column lists the business attributes of the business entity class, while the right column lists the business rules of the business entity. The relative horizontal position of the bubbles represents the hierarchical relationship of the business entity classes. For example, ReservationEntity class 330 is shown below TravelEntity class 310 and above Car class 380 with solid lines connecting their respective bubbles. This spatial arrangement indicates that ReservationEntity class 330 is a child of TravelEntity Class 310 and a parent of Car class 380. Thus, ReservationEntity class 330 inherits all properties of TravelEntity class 310 including all business attributes and business rules definitions. Car class 380, in turn, inherits all properties of ReservationEntity class 330, including all properties of Travel Entity Class 310. Business rules and business attributes of a business entity which are inherited from a different business entity are shown in italics in Fig. 3.

In Fig. 3, RemoteObject entity class 300 is a class of a distributed object system which handles communications over the network, as well as data caching and synchronization of distributed objects. In one embodiment, the RemoteObject entity class is an object-oriented wrapper around an RPC library. However, while the RemoteObject entity class is described for clarity, any other distributed object system known in the art may be used in place of the RemoteObject entity class and the invention is not limited to any specific distributed object system. TravelEntity class 310 is a business entity class which defines the core functions of all business entity classes of Fig. 3, as the remaining business entity classes of Fig. 3 inherit the properties of TravelEntity class 310. TravelEntity class 310 has three business attributes: mode, server and id. The id attribute stores a unique identification string for each instance of TravelEntity class 310, while the server attribute stores the identification string of a server distributed object. The mode field, on the other

hand, is used to keep track of the mode of operation of each instance of TravelEntity class 310, as shown in Figs. 4A and 4B.

In Fig. 4A, mode attribute 400 of business entity A, which is an instance of TravelEntity class 310 (Fig. 3), indicates that business entity A operates in client mode. Alternatively, in Fig. 4B mode attribute 400 of business entity A, which is an instance of TravelEntity class 310 (Fig. 3), indicates that business entity A operates in server mode. Other modes of operations, in addition to server and client modes, can also be used. For example, in the travel reservation example of Fig. 3, five modes of operation are described in the program of Appendix A: client, server, physical, associate and identifier. Different modes are required since in a distributed object environment, multiple copies of a business entity exist on different computers connected to the network. Modes regulate the role of each of the different copies of the business entity with respect to the other copies of the business entity. Initially, a business entity class is instantiated to create a copy of a business entity on a computer to represent a physical object, such as an hotel or a car. That copy of the business entity is thus in physical mode. The business entity is then distributed by creating a remote copy of the business entity (in associate mode) and transmitting the remote copy to a different computer connected to the computer network. The copy of the business entity on the host computer switches from physical mode to server mode to indicate that the business entity has been distributed. If the remote copy of the business entity needs to access the host copy of the business entity, a connection is established over the computer network and the remote copy of the business entity switches from associate mode to client mode for the duration of the connection.

While during certain operations one copy of the business entity acts as a server and another copy of the business entity acts as a client, each copy of the business entity operates in different modes. For example, a copy of a business entity may itself generate a new copy of the business entity and distribute the new copy over the network. In such case the old copy (remote) of the business entity

serves as a server and the new copy (network) serves as a client. Thus, the model of communication among copies of a business entity is more properly characterized as peer-to-peer rather than as client-server.

In addition to business attributes, TravelEntity class 310 also has a number of business rules such as check(), getServiceId(), getType() and getName(). Business rules are also inherited by business entity classes descending from TravelEntity class 310.

ReservationEntity class 330, in turn, provides the functionality common to all business entity classes that can be reserved using the travel reservation system. For example, business rules make() and cancel() are defined in ReservationEntity class 330. Other business entity classes which inherit the properties of TravelEntity class 310 are Flight 320, Warehouse 340, Hotel 350 and Airline 360. Each of these business entity classes defines a set of business attributes and business rules needed to provide functionality to the business entity they represent. For example, business attributes for Flight class 320 include flight, origin, destination and seats, business attributes for Hotel class 350 include name, chain, location and rooms, and business attributes for Airline class 360 include name flights, origins and destinations.

Business entity classes which inherit the properties of ReservationEntity class 330 include Room class 370, Car class 380 and Seat class 390. Room class 370 has business attributes name, category and price. Car class 380 has business attributes license and category. Seat class 390 has business attributes category, price and name. Business rules that provide access to the information contained on each business entity (which is an instance of a business entity class) are defined at various levels of the hierarchy of Fig. 3.

The business entity classes of Fig. 3 are further described by the computer program of Appendix A. In one embodiment, the program of Appendix A is executed on a SunWorkstation II or a SparcStation 10 computer, running under the

SunOS v.4.1.3 operating system, all available from Sun Microsystems Inc. of Mountain View, Calif..

Typically, information on various travel services are maintained by different travel service providers, each utilizing unique data formats. According to some embodiments of the invention, information stored on each merchant's computer system (for example, in a legacy database) is translated into a set of business entities using the adapter, as shown in Fig. 2. The business entities are then distributed and processed over the computer network, as shown in Figs. 5 and 6.

In Fig. 5, multiple copies of the business entities are first created on a host computer in stage 510. The host computer can be either the same computer system on which the source data resides or a different computer connected to the computer on which the source data resides. A copy of the business entity is then stored on the host computer in stage 520. As explained with respect to the mode attribute of TravelEntity 310 (Fig. 3), the copy stored on the host computer represents the actual travel service such as a room in an hotel or a rental car. The remaining copies of the business entity are then distributed to remote computers over the computer network in stage 530. The computer network can be the Internet or any other very large area computer network known in the art. However, the invention is not limited to global or very large area networks, but also includes any computer network such as an intranet, a wide area network or a local area network.

Stage 540 then determines whether the business entity has been deactivated, in which case the operation terminates; otherwise the operation proceeds to stage 550. In stage 550, the copies of the business entities are then processed on either the host computer or one of the remote computers. The remote computers can either process the copies of the business entities directly or translate them back into a specific format (e.g., a legacy database), process them in such a format and translate them back into business entities format through the adapter. Since each copy of a business entity is self-contained, processing of a business entity does not

require communication with the host copy of the business entity. For example, if business entities representing a block of rooms in an hotel have been distributed to a travel agency web site and accessed by a potential customer though a web page maintained by the travel agency, the business entities are able to quote a price and make or cancel a reservation without the need to contact the central reservation system for the hotel chain. Rather, after processing of the copy of the business entity is completed, a connection is established with the host computer and the host copy of the business entity is updated to ensure data consistency in stage 560. Stages 540-560 are then repeated until the business entity is deactivated. This scheme works well if only one copy of a business entity is distributed for each travel service available (e.g. for each hotel room). However, in some situations it may be desirable to distribute multiple copies of a business entity to increase the likelihood of selling the travel service. In these situations, remote copies of the business entities still operate independently (and thus no client-server interaction is required to complete a given operation) but are required to update the other (both host and remote) copies of the business entity within a time window. The time window can be defined in terms of a number of operations, a time period or both.

In some embodiments, distributed business entities contain references to other business entities or to data in formats other than business entities that for efficiency reasons are not distributed along with the copies of the business entities. Processing of business entities under these circumstances is described in Fig. 6. First multiple remote copies of a business entity are created on the host computer in stage 610. These copies contain references to data that is not distributed as part of the business entity, but not copies of the actual data. The host copy of the business entity is then stored on the host computer in stage 620. The remote copies of the business entity are distributed to remote computers over the network in stage 630. Stage 640 then determines whether a remote copy of a business entity contains a reference to data that is required for processing. If the business entity does not contain such a reference, the business entity is processed in stage 650 as

described in stages 540-550 of Fig. 5. Otherwise, the data is retrieved from the host computer over the network in stage 660 and is stored on the remote computer in stage 670.

Furthermore, in some embodiments a distributed business entity can in turn distribute additional (network) copies of the business entity to other (network) computers. For example, in Fig. 2, computer 240 may process business entities A, B and C and determine that business entity B needs to be further processed by computer 250. The remote copy of business entity B can then generate a network copy of business entity B and distribute it to computer 250 over network 200 for further processing. For example, a business entity representing an hotel room may first be distributed to travel agencies over the Internet. However, if after a certain period of time the room has not been booked, a copy of the business entity is generated and sent to a liquidator agency to be sold at discount prices. If a reservation is made either through the travel agency or through the liquidator agency, the copy of the business entity stored on the host computer is updated as described with respect to Fig. 5. When remote copies of a business entity are further distributed to network computers the operation of Figs. 5 and 6 take place between the remote computer and the network computer, rather than between the host computer and the remote computer.

Furthermore, copies of different business entities may interact to perform a certain task. For example, using business entities the travel agency web page described with respect to the prior art would be able to include all the functionality of the web pages maintained by individual travel service providers. In fact, all of the functionality of the individual web pages would be available to the travel agency's computer in the form of business entities. The travel agency's computer could therefore display an integrated web page eliminating the need for a user to follow any links to additional web pages.

Fig. 7 illustrates the structure of a computer system in accordance to one embodiment of the present invention. In Fig. 7, a number of legacy applications

exchange data with business entities through a legacy interface. The business entities, in turn, access the lower layers of the system through an adapter including a commerce platform and facilities. The adapter and the business entity class definitions can be embodied in any computer-readable storage medium, such as a magnetic, a CD-ROM or a carrier wave transmitted over a computer network. The adapter interfaces to other computer systems connected to the network via a series of commercially available interfaces such as CORBA, EDI, MIME and HTML/JAVA. Under these interfaces, sit various communication protocols such as IIOP, SMTP and HTTP. The communication protocols, in turn, communicate with a TCP/IP layer which is to carries out the actual communications between computers connected to the network. Thus, business entities can be accessed using any of the underlying interfaces. For example, a same business entity can be accessed as an HTML web page over the Internet, as a CORBA object over an intranet or as a business entity on the computer on which it is stored.

The distributed business entity system further provides four levels of system security, as shown in Figs. 8A and 8B. In Fig. 8A, a computer 810 is connected over a communication link 830 to a computer 820. Each of computers 810 and 820 has an adapter (Fig. 7). Four types of security checks can be performed on data transmitted back and forth between computers 810 and 820. First, at the network-level encoding and decoding is performed using an SSL library and an RSA key. Second, each computer system performs a second authentication of the message based on an RSA key. Third, at the user-level, access control lists are used to provide control as to which users can access the information transferred over the network. Finally, content-based authentication is performed on each individual message.

Fig. 8B shows a firewall implemented using the distributed business entity system. A third adapter 840 (either on a separate computer or on either one of computers 810 and 820) is inserted between computers 810 and 820. A message transmitted from computer 810 and 820 is thus first transferred over link 850 to

adapter 840 which performs authentication as described with respect to Fig. 8A. If the message satisfies all security checks performed by adapter 840, the message is then forwarded to computer 820 over link 860, otherwise the message is discarded. As a result, adapter 840 provides a firewall 870 between computers 810 and 820.

Embodiments described above illustrate but do not limit the invention. In particular, the invention is not limited to any particular distributed object system. Furthermore, the invention is not limited to any type or number of computers or networks. In fact, the invention can be applied to an arbitrary number of computers connected to an arbitrary number of networks. Therefore, not all computers need to be connected to the same network. Other embodiments and variations are within the scope of the invention, as defined by the following claims.

CLAIMS

We claim:

1. A method of processing business entities over a computer network, wherein each business entity comprises one or more data structures and one or more computer instructions for operating on the data structures, the method comprising:

creating a host copy and a remote copy of a business entity on a host computer connected to the computer network;

storing the host copy of the business entity on the host computer;

transmitting the remote copy of the business entity to a remote computer over the computer network;

performing an operation on the remote copy of the business entity on the remote computer; and

automatically updating the host copy of the business entity in response to the operation on the remote copy of the business entity.

2. The method of Claim 1, further comprising:

performing an operation on the host copy of the business entity on the host computer; and

automatically updating the remote copy of the business entity in response to the operation on the host copy of the business entity.

3. The method of Claim 1, wherein automatically updating a copy of a business entity further comprises:

detecting a disruption of communications between the host and the remote computers over the computer network;

suspending processing of the update until the communications are restored; and

completing the processing of the update once the communications are re-established.

4. The method of Claim 1, wherein each business entity further comprises one or more business attributes, wherein each business attribute includes at least one data structure.

5. The method of Claim 4, wherein updating a copy of a business entity further comprises updating one or more business attributes of the business entity.

6. The method of Claim 1, wherein each business entity further comprises one or more business rules and each business rule includes at least one computer instruction for operating on a data structure.

7. The method of Claim 6, wherein updating a copy of a business entity further comprises updating one or more business rules of the business entity.

8. The method of Claim 1, wherein creating a copy of a business entity further comprises creating an instance of a business entity class.

9. The method of Claim 1, further comprising:

creating a network copy of the business entity on the remote computer connected to the computer network;

transmitting the network copy of the business entity to a network computer over the computer network;

performing an operation on the network copy of the business entity on the network computer; and

automatically updating the remote copy of the business entity in response to the operation on the network copy of the business entity.

10. The method of Claim 9, further comprising:

automatically updating the host copy of the business entity in response to the operation on the network copy of the business entity.

11. The method of Claim 9, further comprising:

performing an operation on the host copy of the business entity on the host computer; and

automatically updating the network copy of the business entity in response to the operation on the host copy of the business entity.

12. A method of processing business entities over a computer network, wherein each business entity comprises one or more data structures and one or more computer instructions for operating on the data structures, the method comprising:

creating a host copy and a remote copy of a business entity on a host computer connected to the computer network;

storing the host copy of the business entity on the host computer;

transmitting the remote copy of the business entity to a remote computer over the computer network;

determining on the remote computer that the remote copy of the business entity contains a reference to data stored on the host computer;

transferring the data from the host computer to the remote computer over the computer network; and

storing the data on the remote computer.

13. A system for processing business entities over a computer network, wherein each business entity comprises one or more data structures and one or more computer instructions for operating on the data structures, the system comprising:

a computer network;

a host computer and a remote computer connected to the computer network;

a host computer program stored on the host computer; and

a remote computer program stored on the remote computer;

wherein the host computer and the remote computer programs comprise instructions for:

creating a host copy and a remote copy of a business entity on the host computer;

storing the host copy of the business entity on the host computer;

transmitting the remote copy of the business entity to the remote computer over the computer network;

determining on the remote computer that the remote copy of the business entity contains a reference to data stored on the host computer;

transferring the data from the host computer to the remote computer over the computer network; and

storing the data on the remote computer.

14. A system for processing business entities over a computer network, wherein each business entity comprises one or more data structures and one or more computer instructions for operating on the data structures, the system comprising:

a computer network;

a host computer and a remote computer connected to the computer network;

a host computer program stored on the host computer; and

a remote computer program stored on the remote computer;

wherein the host computer and the remote computer programs comprise instructions for:

creating a host copy and a remote copy of a business entity on the host computer;

storing the host copy of the business entity on the host computer;

transmitting the remote copy of the business entity to the remote computer over the computer network;

performing an operation on the remote copy of the business entity on the remote computer; and

automatically updating the host copy of the business entity in response to the operation on the remote copy of the business entity.

15. The system of Claim 14, wherein automatically updating the host copy of the business entity further comprises:

detecting a disruption of communications between the host computer and the remote computers over the computer network;

suspending processing of the update until the communications are restored; and

completing the processing of the update once the communications are re-established.

16. The system of Claim 14, wherein each entity comprises one or more business entities and each business entity includes at least one data structure.

17. The system of Claim 14, wherein each business entity comprises one or more business rules and each business rule comprises at least one computer instruction for operating on a data structure.

18. The system of Claim 14, wherein the host computer and the remote computer programs further comprise instructions for:

performing an operation on the host copy of the business entity on the host computer; and

automatically updating the remote copy of the business entity in response to the operation on the host copy of the business entity.

19. The system of Claim 14, wherein the instructions for creating a copy of a business entity further comprise instructions for creating an instance of a business entity.

20. The system of Claim 14, further comprising a network computer program stored on a network computer connected to the computer network, wherein the remote computer and the network computer programs further comprise instructions for:

creating a network copy of the business entity on the remote computer;

transmitting the network copy of the business entity to the network computer over the computer network;

performing an operation on the network copy of the business entity on the network computer; and

automatically updating the remote copy of the business entity in response to the operation on the network copy of the business entity.

21. The system of Claim 20, wherein the network computer program further comprises instructions for:

automatically updating the host copy of the business entity in response to the operation on the network copy of the business entity.

22. A computer-readable storage medium comprising instructions for:

creating a host copy and a remote copy of a business entity on a host computer connected to a computer network;

storing the host copy of the business entity on the host computer;

transmitting the remote copy of the business entity to a remote computer over the computer network;

performing an operation on the remote copy of the business entity on the remote computer; and

automatically updating the host copy of the business entity in response to the operation on the remote copy of the business entity;

wherein each business entity comprises one or more data structures and one or more computer instructions for operating on the data structures.

23. The computer-readable storage medium of Claim 22, further comprising instructions for:

performing an operation on the host copy of the business entity on the host computer; and

automatically updating the remote copy of the business entity in response to the operation on the host copy of the business entity.

24. The computer-readable storage medium of Claim 22, further comprising instructions for creating an instance of a business entity.

25. The computer-readable storage medium of Claim 22, further comprising instructions for:

creating a network copy of the business entity on the remote computer;

transmitting the network copy of the business entity to a network computer over the computer network;

performing an operation on the network copy of the business entity on the network computer; and

automatically updating the remote copy of the business entity in response to the operation on the network copy of the business entity.

26. The computer-readable storage medium of Claim 22, further comprising instructions for:

automatically updating the host copy of the business entity in response to the operation on the network copy of the business entity.

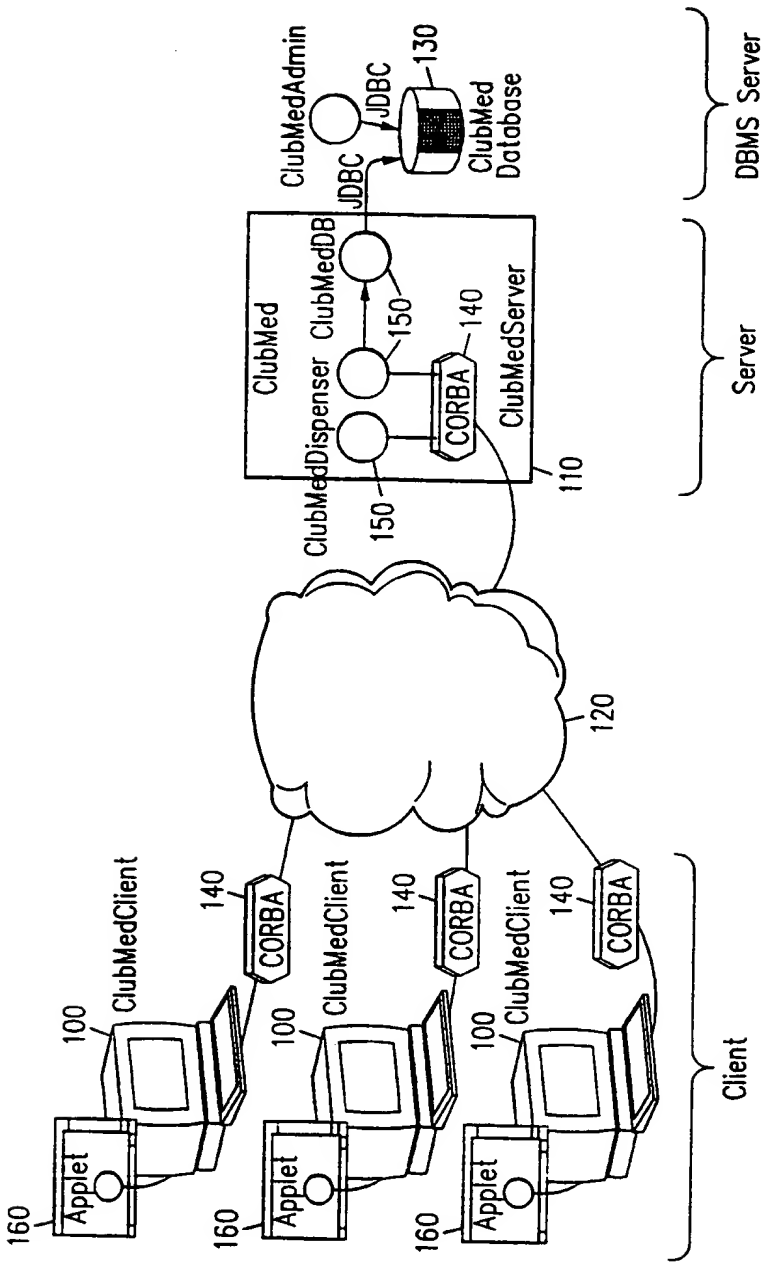


FIG. 1 (PRIOR ART)

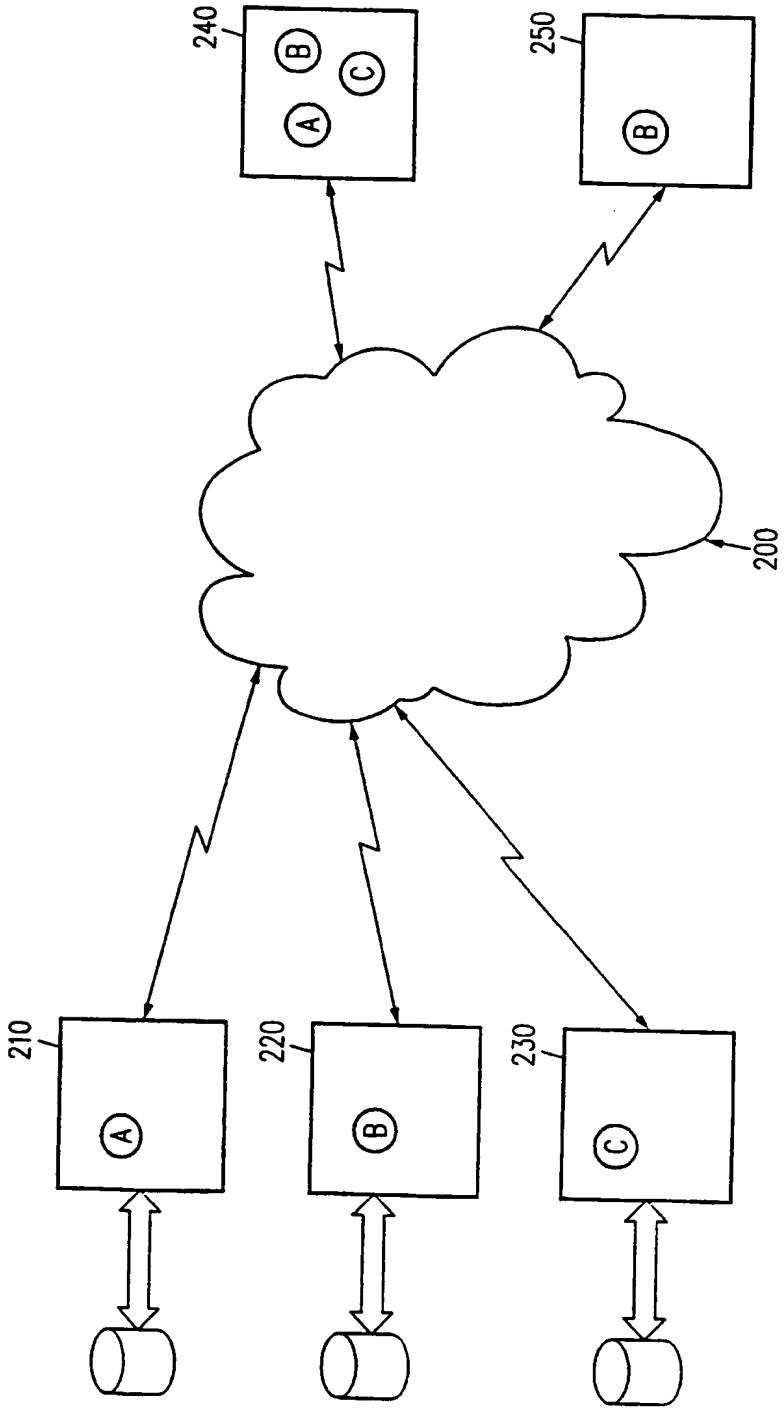


FIG. 2

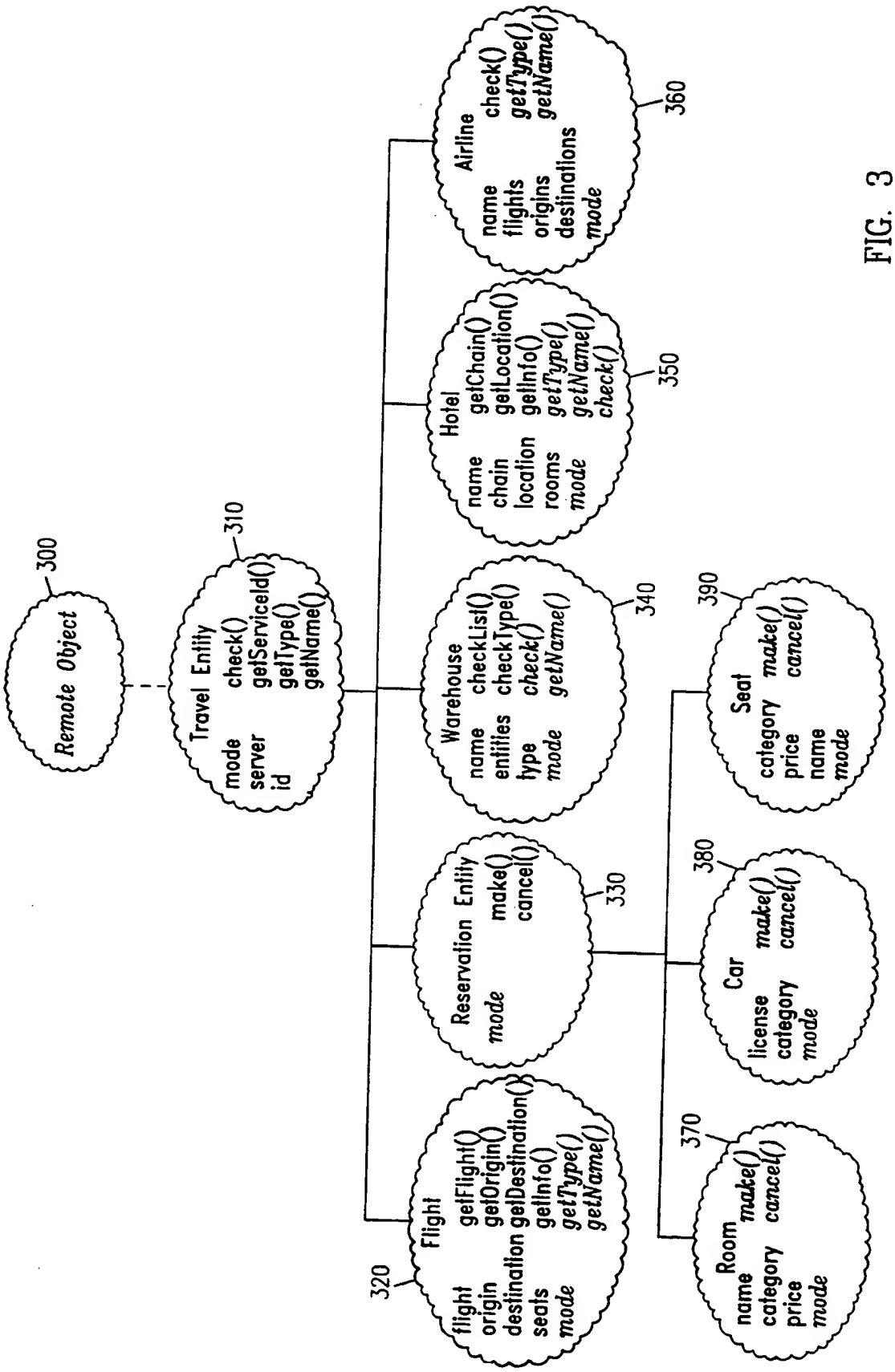

300
Remote Object
310
Travel Entity
mode
server
id
check()
getServiceId()
getType()
getName()
320
Flight
flight
origin
destination
seats
mode
getFlight()
getOrigin()
getDestination()
getInfo()
getType()
getName()
330
Reservation Entity
mode
make()
cancel()
340
Warehouse
name
entities
type
mode
checkList()
checkType()
check()
getName()
350
Hotel
name
chain
location
rooms
mode
getChain()
getLocation()
getInfo()
getType()
getName()
check()
360
Airline
name
flights
origins
destinations
mode
check()
getType()
getName()
370
Room
name
category
price
mode
make()
cancel()
380
Car
license
category
mode
make()
cancel()
390
Seat
category
price
name
mode
make()
cancel()

FIG. 3

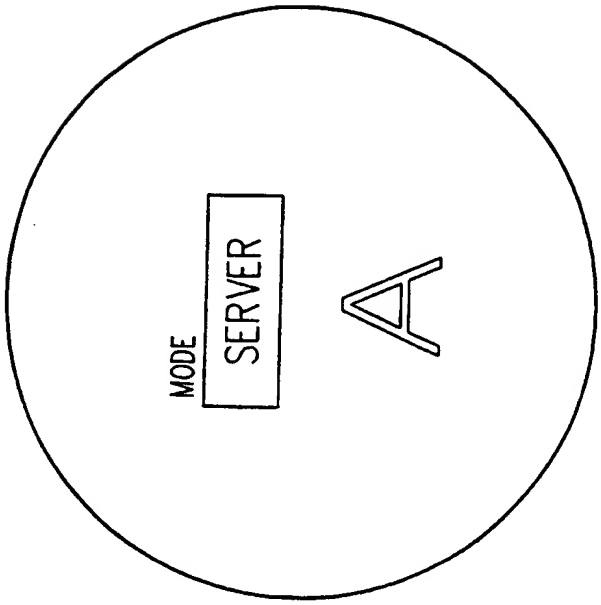


FIG. 4B

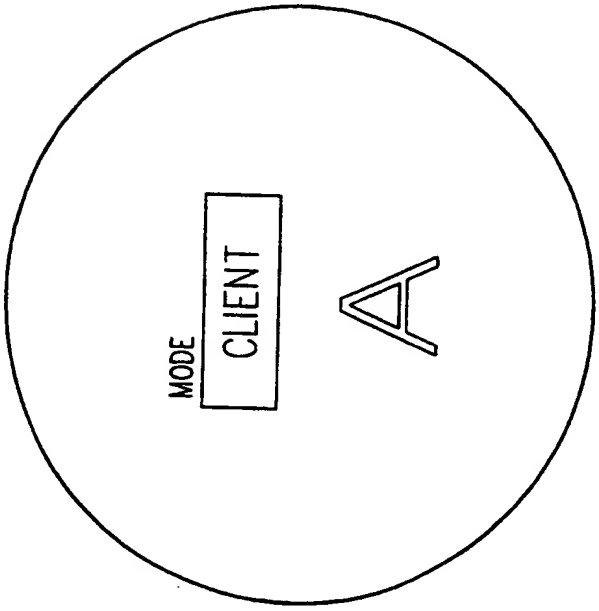


FIG. 4A

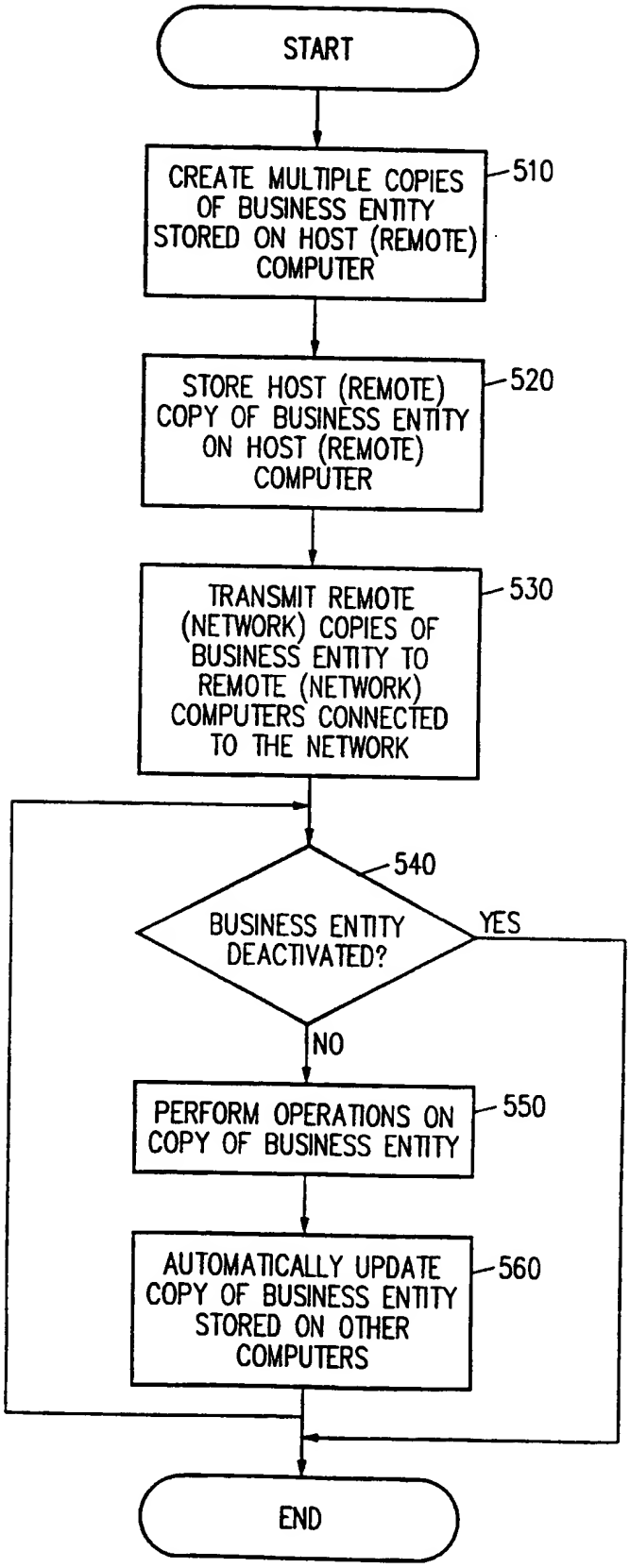

START
CREATE MULTIPLE COPIES OF BUSINESS ENTITY STORED ON HOST (REMOTE) COMPUTER
510
STORE HOST (REMOTE) COPY OF BUSINESS ENTITY ON HOST (REMOTE) COMPUTER
520
TRANSMIT REMOTE (NETWORK) COPIES OF BUSINESS ENTITY TO REMOTE (NETWORK) COMPUTERS CONNECTED TO THE NETWORK
530
540
BUSINESS ENTITY DEACTIVATED?
YES
NO
PERFORM OPERATIONS ON COPY OF BUSINESS ENTITY
550
AUTOMATICALLY UPDATE COPY OF BUSINESS ENTITY STORED ON OTHER COMPUTERS
560
END

FIG. 5

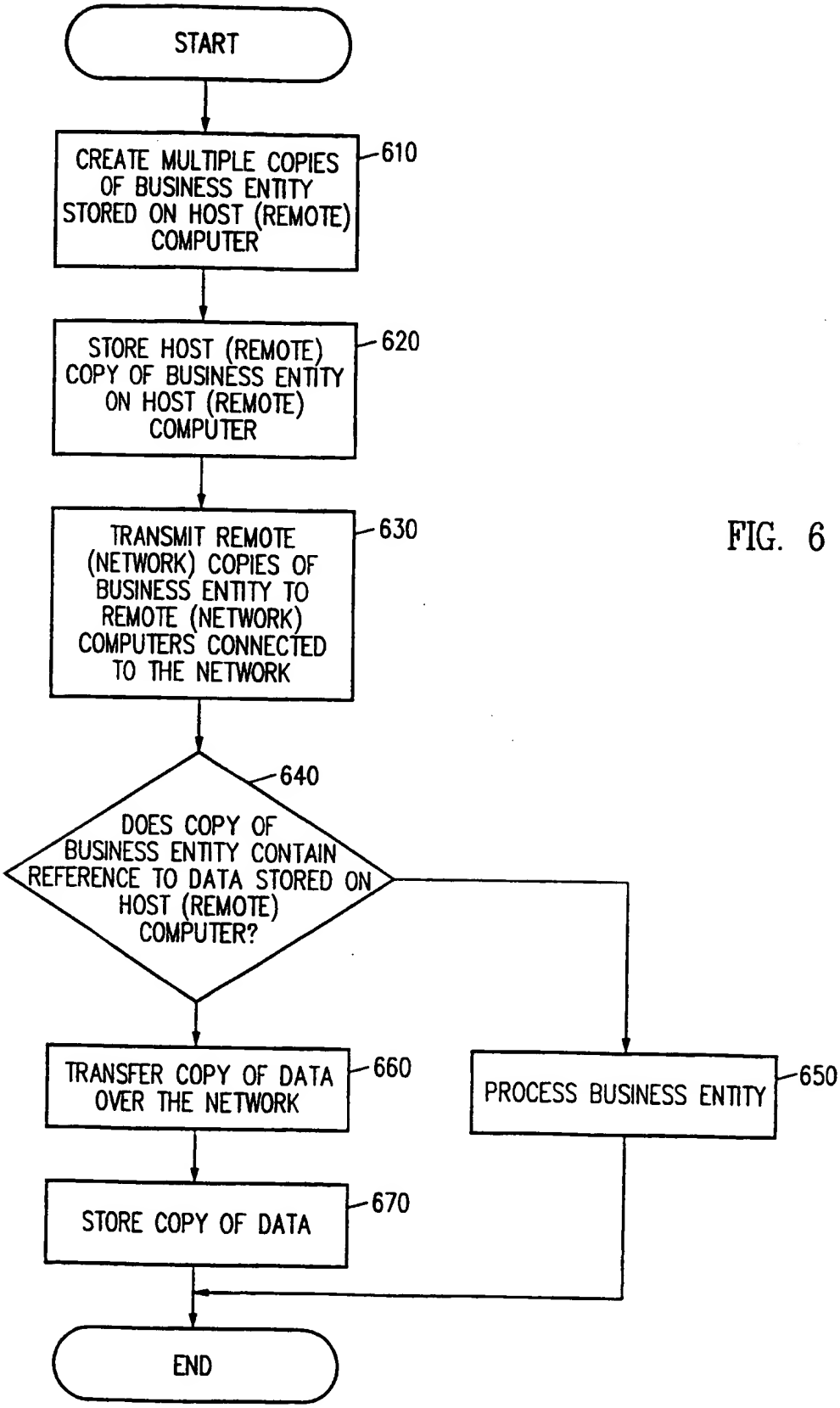

START
CREATE MULTIPLE COPIES OF BUSINESS ENTITY STORED ON HOST (REMOTE) COMPUTER
610
STORE HOST (REMOTE) COPY OF BUSINESS ENTITY ON HOST (REMOTE) COMPUTER
620
TRANSMIT REMOTE (NETWORK) COPIES OF BUSINESS ENTITY TO REMOTE (NETWORK) COMPUTERS CONNECTED TO THE NETWORK
630
640
DOES COPY OF BUSINESS ENTITY CONTAIN REFERENCE TO DATA STORED ON HOST (REMOTE) COMPUTER?
TRANSFER COPY OF DATA OVER THE NETWORK
660
PROCESS BUSINESS ENTITY
650
STORE COPY OF DATA
670
END

FIG. 6

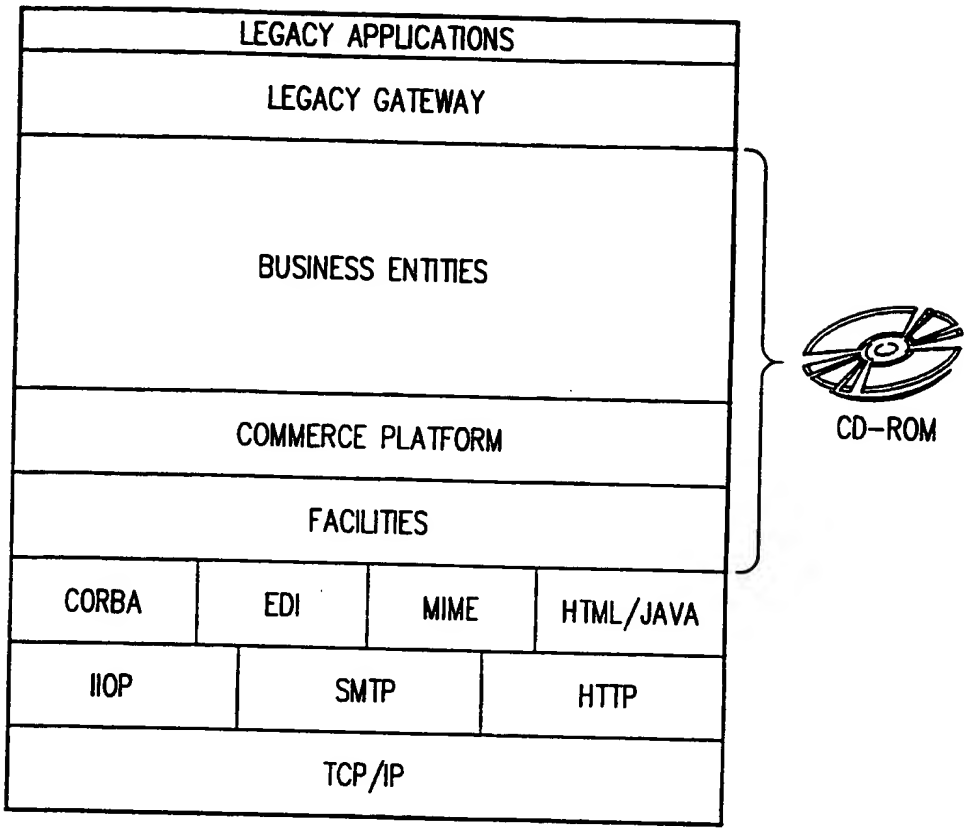


FIG. 7

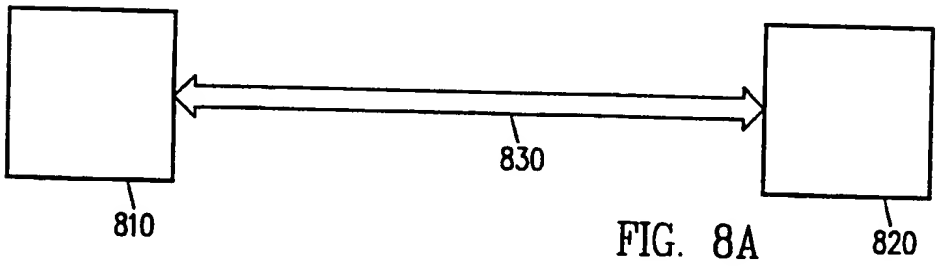


FIG. 8A

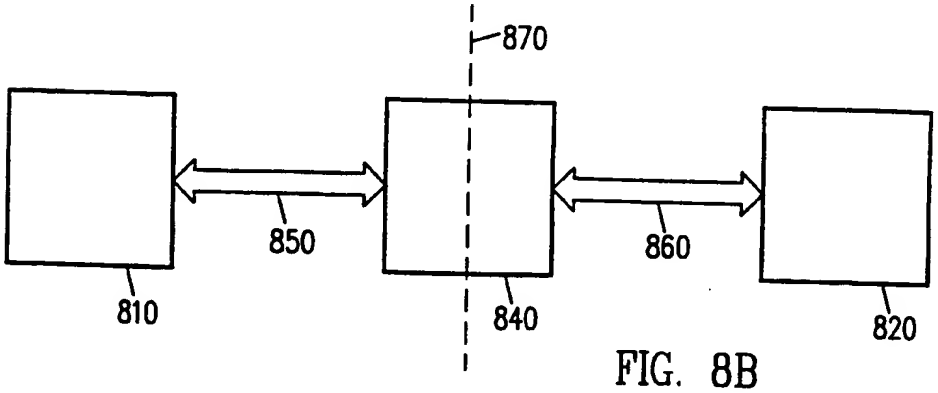


FIG. 8B

INTERNATIONAL SEARCH REPORT

Int tional Application No
PCT/US 98/13585

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X Y A	US 5 245 553 A (TANENBAUM RICHARD) 14 September 1993 see abstract; claims 1-10,15-21; figures 2,9,10 see column 2, line 25 - column 11, line 47	1-3 4-12, 14-21 22-26
X Y A	US 5 107 421 A (YAZIMA KAZUNORI ET AL) 21 April 1992 see abstract; claims 1-5; figures 2-8,10-13 see column 1, line 12 - column 8, line 14	13 14-21 1-12, 22-26

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
1 December 1998	09/12/1998

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Gardiner, A

2

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Int tional Application No
PCT/US 98/13585

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
Y	US 4 788 643 A (TRIPPE KENNETH A B ET AL) 29 November 1988	4-8,12
A	see abstract; claims 1-7; figures 2,3 see column 2, line 61 - column 11, line 45	9-11, 13-26
Y	US 4 769 772 A (DWYER PATRICIA A) 6 September 1988	9-11
A	see abstract; claims 1-18; figures 1-8 see column 1, line 15 - column 5, line 20	2-8, 12-26
P,A	WO 97 27556 A (ELECTRONIC DATA SYST CORP) 31 July 1997 see the whole document	1-26
A	GIFFORD D ET AL: "THE TWA RESERVATION SYSTEM" COMMUNICATIONS OF THE ASSOCIATION FOR COMPUTING MACHINERY, vol. 27, no. 7, 1 July 1984, pages 650-665, XP000563325 see the whole document	1-26

2

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int ıtional Application No
PCT/US 98/13585

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5245553 A	14-09-1993	US 5119319 A	02-06-1992
		AU 2422292 A	02-03-1993
		WO 9303445 A	18-02-1993
		AU 639521 B	29-07-1993
		AU 6972391 A	18-07-1991
		CA 2071951 A	15-06-1991
		EP 0505458 A	30-09-1992
		JP 5502745 T	13-05-1993
		WO 9109378 A	27-06-1991
US 5107421 A	21-04-1992	JP 63289662 A	28-11-1988
US 4788643 A	29-11-1988	NONE	
US 4769772 A	06-09-1988	AU 573423 B	09-06-1988
		AU 5333586 A	04-09-1986
WO 9727556 A	31-07-1997	US 5832451 A	03-11-1998
		AU 1702697 A	20-08-1997
		EP 0876651 A	11-11-1998